UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/598,167	05/16/2007	Michael Luke Tunmer	051035	1921

23696   7590   06/15/2011
QUALCOMM INCORPORATED
5775 MOREHOUSE DR.
SAN DIEGO, CA 92121

EXAMINER
UNG, LANNY N

ART UNIT	PAPER NUMBER
2191	

NOTIFICATION DATE	DELIVERY MODE
06/15/2011	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

us-docketing@qualcomm.com

	Application No.	Applicant(s)
	10/598,167	TUNMER ET AL.
Office Action Summary	Examiner	Art Unit
	LANNY UNG	2191

-- *The MAILING DATE of this communication appears on the cover sheet with the correspondence address* --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
  Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *11 April 2011*.
2a)☒ This action is **FINAL**.   2b)☐ This action is non-final.
3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-7,9-11 and 13-18* is/are pending in the application.
    4a) Of the above claim(s) _____ is/are withdrawn from consideration.
5)☐ Claim(s) _____ is/are allowed.
6)☒ Claim(s) *1-7,9-11 and 13-18* is/are rejected.
7)☐ Claim(s) _____ is/are objected to.
8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.
10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.
    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
    a)☐ All b)☐ Some * c)☐ None of:
       1.☐ Certified copies of the priority documents have been received.
       2.☐ Certified copies of the priority documents have been received in Application No. _____.
       3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08)
    Paper No(s)/Mail Date *February 3, 2011*.
4) ☐ Interview Summary (PTO-413)
    Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____ .

DETAILED ACTION

This Office Action is in response to amendments filed on April 11, 2011.

Claims 1-7, 9-11 and 13-18 are pending.

Claims 1-7, 9-10 and 13-15 have been amended.

Claims 8 and 12 have been canceled.

Claims 16-18 have been added.

Response to Amendment

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for

all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described
> as set forth in section 102 of this title, if the differences between the subject matter sought to
> be patented and the prior art are such that the subject matter as a whole would have been
> obvious at the time the invention was made to a person having ordinary skill in the art to which
> said subject matter pertains. Patentability shall not be negatived by the manner in which the
> invention was made.

Claims 1-7, 9-11, 13-18 are rejected under 35 U.S.C. 103(a) as being

unpatentable over **Edward Bailey** ("Maximum RPM – Taking the RPM Package

Manger to the Limit", 2000) in view of **Yeung et al.** (US 7,496,645) and in further

view of **Marjan Hericko et al.** ("Object Serialization Analysis and Comparison in

Java and .NET", 2003).

With respect to **Claim 1**, Edward Bailey discloses:

a) creating a container, *(creating a package file, Page 21, lines 11-13)* the

container comprising: executable code; *(packages contain programs, Page 21,*

line 14) one or more content resources for use in the user interface; *(packages*

contain documentation, data and configuration information files, Page 21, line 15)

and metadata relating to the one or more content resources, *(every package*

contains information about every file contained in the package, Page 27, lines

33-37)

b) transmitting the container to one or more devices; *(obtain the package*

from an FTP site and stored on a system, Page 32, lines 15-16)

c) extracting the contents of the container at the one or more device; and

(unpacking files from the package and putting them in the proper places in the

system, Page 31, lines 7-11)

d) executing the code to generate a user interface for the one or more

devices. *(performing the installation of the software, Page 31, lines 19-33)*

Edward Bailey does not disclose:

executable code for the user interface

the executable code, the one or more content resources and the metadata

being stored as serialized objects within the container;

However, Yeung et al. disclose:

executable code for the user interface *(software can include high level*

graphical user interface code, Column 1, lines 40-41)

Therefore, it would have been obvious to one of ordinary skill in the art at

the time the invention was made to incorporate the teaching of Yeung et al. into

the teaching of Edward Bailey to include software that contains high level

graphical user interface code in order to install a graphical user interface onto a

system using a package/container.

Edward Bailey and Yeung et al. do not disclose:

the executable code, the one or more content resources and the metadata

being stored as serialized objects within the container;

However, Marjan Hericko et al. disclose:

the executable code, the one or more content resources and the metadata

being stored as serialized objects within the container; *(object serialization is the*

process of writing the state of an object to a stream, Page 44, 1. Introduction,

Paragraph 1, lines 1-6)

Therefore, it would have been obvious to one of ordinary skill in the art at

the time the invention was made to incorporate the teaching of Marjan Hericko et

al. into the teaching of Edward Bailey and Yeung et al. to include storing the

executable code, the content resources and the metadata as serialized objects

within the contain in order to allow for marshaling objects by value and sending

them across process and computer boundaries. *(Marjan Hericko et al., Page 44,*

1. Introduction, Paragraph 1, lines 5-6)

With respect to **Claim 2**, all the limitations of **Claim 1** have been

addressed above; and Edward Bailey further discloses:

wherein the metadata comprise data determining access to the executable

code and/or the one or more content resources to prevent unauthorized access

to the executable code and/or the one or more content resources during step (a).

(every file in the package contains a file's permissions which specify permitted

access, Page 87, lines 15-16)

With respect to **Claim 3**, all the limitations of **Claim 1** have been

addressed above; and <u>Edward Bailey</u> further discloses:

wherein if during step a) the executable code and/or one or more content

resources is altered, the metadata is updated accordingly. *(the MD5 checksum of*

a file will change if there is any change to the file's contents, Page 54 and 55,

lines 46-48 and 1-2 respectively)

With respect to **Claim 4**, all the limitations of **Claim 1** have been

addressed above; and <u>Edward Bailey</u> further discloses:

the metadata relating to the one or more content resources relates to one

or more hierarchical classification(s), the hierarchical classification(s) relating to

the capabilities of the device. *(every file contained in the package includes*

information about the file size, Page 87, lines 31-33)

With respect to **Claim 5**, all the limitations of **Claim 1** have been

addressed above; and <u>Marjan Hericko et al.</u> further discloses:

further comprising the step of e) processing the container contents into a

format for transmission to the device, step e) being performed subsequent 5 to

step a) and prior to step b). *(the process of writing the state of an object to a*

stream, object serialization, to allow for sending the object across process and

computer boundaries, Page 44, 1. Introduction, Paragraph 1, lines 1-6)

With respect to **Claim 6**, Edward Bailey discloses:

storage means to receive a data container; *(disk drive, Page 20, lines 14-15)*

editing means to enable the data container to be edited, *(creating a package file, Page 21, lines 11-13)* the data container comprising executable code; *(packages contain programs, Page 21, line 14)* one or more content resources for use in the user interface; *(packages contain documentation, data and configuration information files, Page 21, line 15)* and metadata relating to the one or more content resources, *(every package contains information about every file contained in the package, Page 27, lines 33-37)*

and transmission means for transmitting a data container to one or more devices. *(obtain the package from an FTP site and stored on a system, Page 32, lines 15-16)*

Edward Bailey does not disclose:

the data container comprising executable code for the user interface

the executable code, the one or more content resources and the metadata being stored as serialized objects within the data container;

However, Yeung et al. disclose:

the data container comprising executable code for the user interface

(software can include high level graphical user interface code, Column 1, lines 40-41)

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of <u>Yeung et al.</u> into the teaching of <u>Edward Bailey</u> to include software that contains high level graphical user interface code in order to install a graphical user interface onto a system using a package/container.

<u>Edward Bailey</u> and <u>Yeung et al.</u> do not disclose:

the executable code, the one or more content resources and the metadata being stored as serialized objects within the data container;

However, <u>Marjan Hericko et al.</u> disclose:

the executable code, the one or more content resources and the metadata being stored as serialized objects within the data container; *(objects serialization, Page 44, 1. Introduction, Paragraph 1, lines 1-6)*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of <u>Marjan Hericko et al.</u> into the teaching of <u>Edward Bailey</u> and <u>Yeung et al.</u> to include storing the executable code, the content resources and the metadata as serialized objects within the contain in order to allow for marshaling objects by value and sending them across process and computer boundaries. *(<u>Marjan Hericko et al.</u>, Page 44, 1. Introduction, Paragraph 1, lines 5-6)*

With respect to **Claim 7**, all the limitations of **Claim 6** have been addressed above; and <u>Marjan Hericko et al.</u> further discloses:

wherein the server further comprises a processing means configured, in use, to process the data container prior to transmission of the data container to one or more devices. *(the process of writing the state of an object to a stream, object serialization, to allow for sending the object across process and computer boundaries, Page 44, 1. Introduction, Paragraph 1, lines 1-6)*

With respect to **Claim 9**, <u>Edward Bailey</u> discloses:

a) receiving at the device a container over a communications network, *(obtain the package from an FTP site and stored on a system, Page 32, lines 15-16)* the container comprising: executable code; *(packages contain programs, Page 21, line 14)* one or more content resources for use in the user interface; *(packages contain documentation, data and configuration information files, Page 21, line 15)* and metadata relating to the one or more content resources, *(every package contains information about every file contained in the package, Page 27, lines 33-37)*

b) extracting the contents of the container at the device; and *(unpacking files from the package and putting them in the proper places in the system, Page 31, lines 7-11)*

c) executing the code to generate a user interface for the device. *(performing the installation of the software, Page 31, lines 19-33)*

<u>Edward Bailey</u> does not disclose:

executable code for the user interface

the executable code, the one or more content resources and the metadata

being stored as serialized objects within the container;

However, Yeung et al. disclose:

executable code for the user interface *(software can include high level*

graphical user interface code, Column 1, lines 40-41)

Therefore, it would have been obvious to one of ordinary skill in the art at

the time the invention was made to incorporate the teaching of Yeung et al. into

the teaching of Edward Bailey to include software that contains high level

graphical user interface code in order to install a graphical user interface onto a

system using a package/container.

Edward Bailey and Yeung et al. do not disclose:

the executable code, the one or more content resources and the metadata

being stored as serialized objects within the container;

However, Marjan Hericko et al. disclose:

the executable code, the one or more content resources and the metadata

being stored as serialized objects within the container; *(objects serialization,*

Page 44, 1. Introduction, Paragraph 1, lines 1-6)

Therefore, it would have been obvious to one of ordinary skill in the art at

the time the invention was made to incorporate the teaching of Marjan Hericko et

al. into the teaching of Edward Bailey and Yeung et al. to include storing the

executable code, the content resources and the metadata as serialized objects

within the contain in order to allow for marshaling objects by value and sending

them across process and computer boundaries. *(Marjan Hericko et al., Page 44,*

1. Introduction, Paragraph 1, lines 5-6)

With respect to **Claim 10**, all the limitations of **Claim 9** have been

addressed above; and <u>Edward Bailey</u> further discloses:

wherein the metadata comprises data determining access to the

executable code and/or the one or more content resources to control access to

the executable code and/or the one or more content resources during step (b).

(every file in the package contains a file's permissions which specify permitted

access, Page 87, lines 15-16)

With respect to **Claim 11**, all the limitations of **Claim 10** have been

addressed above; and <u>Edward Bailey</u> further discloses:

wherein the access- determining metadata can be updated in response to

receiving a control message from the communications network. *(the MD5*

checksum of a file will change if there is any change to the file's contents, Page

54 and 55, lines 46-48 and 1-2 respectively)

With respect to **Claim 13**, <u>Edward Bailey</u> discloses:

receive a data container from a communications network via the

communications interface; *(obtain the package from an FTP site and stored on a*

system, Page 32, lines 15-16)

store the data container in the storage means; *(mass storage device, i.e. disk drive, Page 20, lines 11-12)*

process the data container using the processing means to extract the contents of the data container, *(unpacking files from the package and putting them in the proper places in the system, Page 31, lines 7-11)* the data container comprising executable code; *(packages contain programs, Page 21, line 14)* one or more content resources for use in the user interface; *(packages contain documentation, data and configuration information files, Page 21, line 15)* and metadata relating to the one or more content resources, *(every package contains information about every file contained in the package, Page 27, lines 33-37)*

Edward Bailey does not disclose:

executable code for the user interface

the executable code, the one or more content resources and the metadata being stored as serialized objects within the data container;

form a user interface in accordance with the extracted contents of the data container;

and display the user interface in the device display.

However, Yeung et al. disclose:

executable code for the user interface *(software can include high level graphical user interface code, Column 1, lines 40-41)*

form a user interface in accordance with the extracted contents of the data container; *(inherent the high level graphical user interface code when executed produces the user interface, Column 1, lines 40-41)*

and display the user interface in the device display. *(inherent the user interface code is displayed on a display device, Column 1, lines 40-42)*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Yeung et al. into the teaching of Edward Bailey to include software that contains high level graphical user interface code in order to install a graphical user interface onto a system using a package/container.

Edward Bailey and Yeung et al. do not disclose:

the executable code, the one or more content resources and the metadata being stored as serialized objects within the data container;

However, Marjan Hericko et al. disclose:

the executable code, the one or more content resources and the metadata being stored as serialized objects within the data container; *(objects serialization, Page 44, 1. Introduction, Paragraph 1, lines 1-6)*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Marjan Hericko et al. into the teaching of Edward Bailey and Yeung et al. to include storing the executable code, the content resources and the metadata as serialized objects within the contain in order to allow for marshaling objects by value and sending them across process and computer boundaries. *(Marjan Hericko et al., Page 44, 1. Introduction, Paragraph 1, lines 5-6)*

With respect to **Claim 14**, all the limitations of **Claim 13** have been

addressed above; and <u>Edward Bailey</u> further discloses:

wherein the metadata stored in the storage means comprises data

determining access to the executable code and/or the one or more content

resources to control access to the executable code and/or the one or more

content resources. *(every file in the package contains a file's permissions which*

specify permitted access, Page 87, lines 15-16)

With respect to **Claim 15**, all the limitations of **Claim 14** have been

addressed above; and <u>Edward Bailey</u> further discloses:

wherein the device is further configure, in use, to receive control

commands from the communications network via the communications interface,

the control commands updating the metadata that determines access to the

executable code and/or the one or more content resources. *(the MD5 checksum*

of a file will change if there is any change to the file's contents, Page 54 and 55,

lines 46-48 and 1-2 respectively)

With respect to **Claim 16**, all the limitations of **Claim 1** have been

addressed above; and <u>Edward Bailey</u> further discloses:

wherein the metadata is updated in an update packet *(the MD5 checksum*

(metadata) of a file will change if there is any change to the file's contents, Page

54 and 55, lines 46-48 and 1-2 respectively)

<u>Edward Bailey</u> does not disclose:

an update packet defined as a binary serialization of an XML schema

However, <u>Marjan Hericko et al.</u> disclose:

an update packet defined as a binary serialization of an XML schema

(serialization can represent the serialized state of an object using binary format,

Page 44, 1. Introduction, Paragraph 2, lines 1-3)

Therefore, it would have been obvious to one of ordinary skill in the art at

the time the invention was made to incorporate the teaching of <u>Marjan Hericko et</u>

<u>al.</u> into the teaching of <u>Edward Bailey</u> to include using binary serialization for the

update packet format in order to take advantage of the benefits of binary

serialization such as being more effective in terms of memory space and time, as

it needs to do much less transformations than XML serialization. *(Marjan Hericko*

et al., 1. Introduction, Paragraph 2, lines 1-3)

With respect to **Claim 17**, all the limitations of **Claim 2** have been

addressed above; and <u>Edward Bailey</u> further discloses:

wherein if during step a) the executable code and/or one or more content

resources is altered, the metadata is updated accordingly. *(the MD5 checksum of*

a file will change if there is any change to the file's contents, Page 54 and 55,

lines 46-48 and 1-2 respectively)

With respect to **Claim 18**, all the limitations of **Claim 17** have been

addressed above; and <u>Edward Bailey</u> further discloses:

wherein the metadata is updated in an update packet *(the MD5 checksum (metadata) of a file will change if there is any change to the file's contents, Page 54 and 55, lines 46-48 and 1-2 respectively)*

Edward Bailey does not disclose:

an update packet defined as a binary serialization of an XML schema

However, Marjan Hericko et al. disclose:

an update packet defined as a binary serialization of an XML schema *(serialization can represent the serialized state of an object using binary format, Page 44, 1. Introduction, Paragraph 2, lines 1-3)*

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Marjan Hericko et al. into the teaching of Edward Bailey to include using binary serialization for the update packet format in order to take advantage of the benefits of binary serialization such as being more effective in terms of memory space and time, as it needs to do much less transformations than XML serialization. *(Marjan Hericko et al., 1. Introduction, Paragraph 2, lines 1-3)*

Response to Arguments

Applicant's arguments filed April 11, 2011 have been fully considered but they are not persuasive.

In the Remarks, Applicant argues:

Applicants disagree with the Office Action's characterization of the
teachings of Hericko. Applicants submit that nothing in the above cited portion of
Hericko, or in the remainder of Hericko teaches or suggests the serialization or
storage of executable code, content resources, or metadata as objects'. Rather,
Hericko simply documents the results of an evaluation of the performance of
serialization mechanisms with respect to the memory space used for a
serialization process. See Hericko, page 44, section 2, paragraph 1, lines 1-3.
Therefore, Applicants submit that Hericko is silent regarding the element of the
executable code, the one or more content resources and the metadata being
stored as serialized objects within the container." Thus, none of Bailey, Yeung,
and Hericko, alone or in combination, teach or suggest all of the elements recited
in amended claims 1, 6, 9, and 13.

Examiner's response:

The Examiner respectfully disagrees. The primary reference, Bailey, was
used to disclose the various elements of "executable code... one or more content
resources... metadata relating to one ore more content resources" (see the
rejection of Claims 1, 6, 9 and 13) while the Hericko reference was used to teach
the concept of storing objects as serialized objects (object serialization) and why
it would be beneficial to do so. Hericko discloses that object serialization is a
process of "writing the state of an object to a stream" which "allows marshaling
objects by value and sending them across process and computer boundaries".
(see Page 44, 1. Introduction, Paragraph 1, lines 1-6) Storing objects as

serialized objects, i.e. object serialization (as claimed by applicant in the various independent claims) is well known within the art according to the Hericko reference, which has a date of August 2003.

Therefore, for at least the reasons set forth above, the rejections made under 35 U.S.C. §103 with respect to Claims 1, 6, 9 and 13 are proper, and therefore, maintained.

In the Remarks, Applicant argues:

Applicants submit that the asserted combination of Bailey, Yeung, and Hericko is improper and insufficient to render the claims prima facie obvious because the proposed modification of the prior art changes the principle of operation of the prior art invention being modified in violation of M.P.E.P. § 2143.01. See In re Ratti, 270 F.2d 810, 123 USPQ 349 (CCPA 1959). Yeung specifically operates to update a graphical user interface by not requiring transferred software to be in a packaged format. See Yeung Col. 3, lines 24-27. The invention in Yeung operates in an unpackaged format to provide a "website owner greater flexibility in its choice of tools to employ for the transfer process." Id. Therefore, any modification of Yeung to provide a container comprising executable code, one or more content resources, and metadata relating to the one or more content resources would impermissibly change the principle of operation of Yeung. Thus, the combination of Yeung with Bailey and Hericko is improper.

As admitted in the Office Action, "Edward Bailey does not disclose: executable code for a user interface." Office Action dated January 18, 2011, page 4. Applicants submit that Hericko is silent with respect to executable code for a user interface. The Office Action cited Yeung to cure the deficiencies of Bailey (and Hericko) to provide a teaching of an executable code for a user interface.

Applicants submit that since it is improper to combine Yeung with Bailey and Hericko to arrive at the invention recited in the claims, the Office Action fails to establish a prima facie case of obviousness with respect to claims 1, 6, 9, and 13.

Examiner's response:

The Examiner respectfully disagrees. The Yeung reference was used to simply teach that executable code can be user interface code. (*Yeung*, *Column 1, lines 40-41*) The primary reference, Bailey, teaches executable code (*Bailey*, *Page 21, line 14*) but is silent on specifically what type of code this is. Therefore, the Examiner has used the Yeung reference to disclose that the Bailey executable code can be specifically user interface code. It is believed that the applicant has misinterpreted the Examiner's combination. The Examiner has not modified Yeung's invention to include "a container comprising executable code, one ore more content resources, and metadata relating to one or more content resources" but has modified the Bailey reference to include executable code that is specifically user interface code (provided from the Yeung reference).

Therefore, for at least the reasons set forth above, the rejections made

under 35 U.S.C. §103 with respect to Claims 1, 6, 9 and 13 are proper, and

therefore, maintained.

Conclusion

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of

time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire

THREE MONTHS from the mailing date of this action. In the event a first reply is

filed within TWO MONTHS of the mailing date of this final action and the advisory

action is not mailed until after the end of the THREE-MONTH shortened statutory

period, then the shortened statutory period will expire on the date the advisory

action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be

calculated from the mailing date of the advisory action. In no event, however, will

the statutory period for reply expire later than SIX MONTHS from the mailing

date of this final action.

Any inquiry concerning this communication or earlier communications from

the examiner should be directed to LANNY UNG whose telephone number is

(571)270-7708. The examiner can normally be reached on Monday-Thursday,

8:am-5:30pm EST.

If attempts to reach the examiner by telephone are unsuccessful, the

examiner's supervisor, Wei Zhen can be reached on (571)272-3708. The fax

phone number for the organization where this application or proceeding is

assigned is 571-273-8300.

Information regarding the status of an application may be obtained from

the Patent Application Information Retrieval (PAIR) system. Status information

for published applications may be obtained from either Private PAIR or Public

PAIR. Status information for unpublished applications is available through

Private PAIR only. For more information about the PAIR system, see http://pair-

direct.uspto.gov. Should you have questions on access to the Private PAIR

system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-

free). If you would like assistance from a USPTO Customer Service

Representative or access to the automated information system, call 800-786-

9199 (IN USA OR CANADA) or 571-272-1000.

/LU/
Lanny Ung
Examiner, Art Unit 2191
June 8, 2011
/WEI ZHEN/
Supervisory Patent Examiner, Art Unit 2191